



M.I.T. LIBRARIES - DEWEY

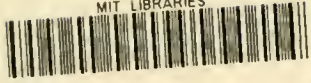
HD28
.M414

no. 3352
-91

45



MIT LIBRARIES



3 9080 00748710 8

**IMPROVING THE RELIABILITY
OF FUNCTION POINT MEASUREMENT:
AN EMPIRICAL STUDY**

**Chris F. Kemerer
Benjamin S. Porter**

October 1991

**CISR WP No. 229
Sloan WP No. 3352-91-MSA**

Center for Information Systems Research

Massachusetts Institute of Technology
Sloan School of Management
77 Massachusetts Avenue
Cambridge, Massachusetts, 02139



**IMPROVING THE RELIABILITY
OF FUNCTION POINT MEASUREMENT:
AN EMPIRICAL STUDY**

**Chris F. Kemerer
Benjamin S. Porter**

October 1991

**CISR WP No. 229
Sloan WP No. 3352-91-MSA**

©1991 C.F. Kemerer, B.S. Porter

**Center for Information Systems Research
Sloan School of Management
Massachusetts Institute of Technology**

M.I.T. LIBRARIES
FEB 21 1992
RECEIVED

Improving the Reliability of Function Point Measurement:

An Empirical Study

Chris F. Kemerer

**Massachusetts Institute of Technology
E53-315**

**50 Memorial Drive
Cambridge, MA 02139
ckemerer@sloan.mit.edu
617/253-2971 (o)
617/258-7579 (fax)**

Benjamin S. Porter

**DMR Group, Inc.
12 Post Office Square
Boston, MA 02109
617/451-9500 (o)
617/695-1537 (fax)**

October 1991

Research support from the International Function Point Users Group and MIT's Center for Information Systems Research is gratefully acknowledged. The cooperation of A. Belden, M. Braun, and J. Frisbie was invaluable in providing data for this research. Helpful comments were received from J.M. Desharnais, F. Mazzucco, J. Quillard, R. Selby, C. Scates, W. Rumpf, and L. Smith on an earlier version.

Improving the Reliability of Function Point Measurement:

An Empirical Study

ABSTRACT

Information Systems development has operated for virtually its entire history without the quantitative measurement capability of other business functional areas such as marketing or manufacturing. Today, managers of Information Systems organizations are increasingly taken to task to measure and report, in quantitative terms, the effectiveness and efficiency of their internal operations. In addition, measurement of information systems development products is also an issue of increasing importance due to the growing costs associated with information systems development and maintenance.

One measure of the size and complexity of information systems that is growing in acceptance and adoption is Function Points, a user-oriented non-source line of code metric of the product of systems development. Recent previous research has documented the degree of reliability of Function Points as a metric. This research extends that work by (a) identifying the major sources of variation through a survey of current practice, and (b) estimating the magnitude of the effect of these sources of variation using detailed case study data from actual commercial systems.

The results of the research show that a relatively small number of factors have the greatest potential for affecting reliability, and recommendations are made for using these results to improve the reliability of Function Point counting in organizations.

ACM CR Categories and Subject Descriptors: D.2.8 (Software Engineering): Metrics; D.2.9 (Software Engineering): Management; K.6.0 (Management of Computing and Information Systems): General - *Economics*; K.6.1 (Management of Computing and Information Systems): Project and People Management; K.6.3 (Management of Computing and Information Systems): Software Management

General Terms: Management, Measurement, Performance, Estimation, Reliability.

Additional Key Words and Phrases: Function Points, Project Planning, Productivity Evaluation.

1. INTRODUCTION

Management of software development and maintenance encompasses two major functions, planning and control, both of which require the capability to accurately and reliably measure the software being delivered. *Planning* of software development projects emphasizes estimation of the size of the delivered system in order that appropriate budgets and schedules can be agreed upon. Without valid size estimates, this process is likely to be highly inaccurate, leading to software that is delivered late and over-budget. *Control* of software development requires a means to measure progress on the project and to perform after-the-fact evaluations of the project in order, for example, to evaluate the effectiveness of the tools and techniques employed on the project to improve productivity and quality.

Unfortunately, as current practice often demonstrates, both of these activities are typically not well performed, in part because of the lack of well-accepted measures, or metrics.

Software size is a critical component of productivity and quality ratios, and has traditionally been measured by the number of source lines of code (SLOC) delivered in the final system. This metric has been criticized in both its planning and control applications. In planning, the task of estimating the final SLOC count for a proposed system has been shown to be difficult to do accurately in actual practice (Low and Jeffery 1990). And in control, SLOC measures for evaluating productivity have weaknesses as well, in particular, the problem of comparing systems written in different languages (Jones 1986).

Against this background, an alternative software size metric was developed by Allan Albrecht of IBM (Albrecht and Gaffney 1983). This metric, which he termed "function points" (hereafter FPs), is designed to size a system in terms of its delivered functionality, measured as a weighted sum of numbers of inputs, outputs, inquiries, and files. Albrecht argued that these components would be much easier to estimate than SLOC early in the software project life-cycle, and would be generally more meaningful to non-programmers.

In addition, for evaluation purposes, they would avoid the difficulties involved in comparing SLOC counts for systems written in different languages.

FPs have proven to be a broadly accepted metric with both practitioners and academic researchers. Dreger estimates that some 500 major corporations world-wide are using FPs (Dreger 1989), and, in a survey by the Quality Assurance Institute, FPs were found to be regarded as the best available MIS productivity metric (Perry 1986). They have also been widely used by researchers in such applications as cost estimation (Kemerer 1987), software development productivity evaluation (Behrens 1983) (Rudolph 1983), software maintenance productivity evaluation (Banker *et al.* 1991), software quality evaluation (Cooprider and Henderson 1989) and software project sizing (Banker and Kemerer 1989). Additional work in defining standards has been done by Zwanzig (Zwanzig 1984) and Desharnais (Desharnais 1988). Although originally developed by Albrecht for traditional MIS applications, recently there has been significant work in extending FPs to scientific and real time systems (Jones 1988; Reifer 1990; Whitmire *et al.* 1991).

Despite their wide use by researchers and their growing acceptance in practice, FPs are not without criticism. The main criticism revolves around the alleged low inter-rater reliability of FP counts, that is, whether two individuals performing a FP count for the same system would generate the same result (Carmines and Zeller 1979). Barry Boehm, a leading researcher in the software estimation and modeling area, has described the definitions of function types as "ambiguous" (Boehm 1987). And, the author of a leading software engineering textbook summarizes his discussion of FPs as follows:

"The function-point metric, like LOC, is relatively controversial...Opponents claim that the method requires some 'sleight of hand' in that computation is based on subjective, rather than objective, data..." (Pressman 1987, p. 94)

This perception of FPs as being unreliable has undoubtedly slowed their acceptance as a metric, as both practitioners and researchers may feel that in order to ensure sufficient measurement reliability either a) a single individual would be required to count all

systems, or b) multiple raters should be used for all systems and their counts averaged to approximate the 'true' value (Symons 1988). Both of these options are unattractive in terms of either decreased flexibility in the first case and likely increased cost and cycle times in the second.

Against this background some recent research has measured the actual magnitude of the inter-rater reliability. Kemerer performed a field experiment where pairs of systems developers measured FP counts for completed medium-sized commercial systems (Kemerer 1991). The results of this analysis were that the pairs of FP counts were highly correlated ($\rho = .8$) and had an average variance of approximately eleven percent.

While these results are encouraging for the continued use of FPs, as the reliability is much higher than previously speculated, there is clearly still room for improvement. In particular, given that one use of FPs is for managerial control in the form of post-implementation productivity and quality evaluations, an 11% variance in counting could mask small but real underlying productivity changes, and therefore could interfere with proper managerial decision making. For example, a software project might have been a pilot test for use of a new tool or method, which resulted in a ten percent productivity gain. If, through unfortunate coincidence the output of this project was understated by eleven percent, then managers might come to the mistaken conclusion that the new tool or method had no or even a slightly negative impact, and thus inappropriately abandon it.

Given this and similar scenarios, it is clearly important for management to have reliable instruments with which to measure their output. And, given that (1) FPs are already widely in use as a metric, and (2) have been shown to have good but imperfect reliability, it seems appropriate to attempt to determine the sources of the variation in counting as a first step towards eliminating them and making FPs an even more reliable metric.

The previous research described above used a large scale experimental design to identify the magnitude of the variations in FP counting. However, that research approach is ill-

suited to the detailed analysis necessary to address the *source* of the variations in reliability. Therefore, this paper reports on the results of a two-phased research approach that is complementary to the research described earlier. The first phase used a combination of key informants and a field survey to identify the most likely sources of FP counting variance. The second phase collected data from three detailed case studies which were then used to estimate the magnitude of effect of the variations. In all, thirty-three FP counts were estimated from the detailed case study data.

The results from this analysis identified three potential sources of variation in FP counting: the treatment of backup files, menus, and external files used as transactions. These are the three areas where tighter standards are necessary and where managers should focus their attention on adopting and adhering to standard counting practices. The results of this research also identified several areas that have been suggested to cause variation, but may not be important sources of error in actual practice. These include treatment of error message responses and hard coded tables.

This paper is organized as follows. Section 2 presents a brief description of the research problem and the previous research. Section 3 describes the research methodology, which consisted of a survey and a set of quantitative case studies. Results of this analysis are presented in Section 4, and Section 5 offers some concluding remarks.

2. RESEARCH PROBLEM

2.1. Introduction

The uses of software measurement are as varied as the organizations which are putting the measures into practice. One widespread use of software measurement is to improve the estimation of the size of development projects. Much of the early literature on software measurement focuses on the complexities of estimation (Boehm 1981) (Jones 1986).

It has only been within the past several years that many organizations have begun systematically collecting a wide variety of data about their software development and maintenance activities. These measurement activities are the advent of both management programs (designed to set and achieve various effectiveness and efficiency objectives) and professional development programs (assisting professionals in the furtherance of their development and maintenance skills).

2.2. Previous Research

Despite both the widespread use of FPs and some attendant criticism of their suspected lack of reliability, there has been little research on this question. Perhaps the first attempt at investigating the inter-rater reliability question was made by members of the IBM GUIDE Productivity Project Group, the results of which are described by Rudolph as follows:

“In a pilot experiment conducted in February 1983 by members of the GUIDE Productivity Project Group ...about 20 individuals judged independently the function point value of a system, using the requirement specifications. Values within the range $\pm 30\%$ of the average judgement were observed ...The difference resulted largely from differing interpretation of the requirement specification. This should be the upper limit of the error range of the function point technique. Programs available in source code or with detailed design specification should have an error of less than $\pm 10\%$ in their function point assessment. With a detailed description of the system there is not much room for different interpretations.” (Rudolph 1983, p. 6)

Aside from this description, the only other research documented study is by Low and Jeffery (Low and Jeffery 1990). Their research focused on the inter-rater reliability of FP counts using as their research methodology an experiment using professional systems developers as subjects, with the unit of analysis being a set of program level specifications. Two sets of program specifications were used, both pre-tested with student subjects. For the inter-rater reliability question, 22 systems development professionals who counted FPs as part of their employment in seven Australian organizations were used, as were an additional 20 inexperienced raters who were given training in the then current Albrecht standard. Each of the experienced raters used his or her organization's own variation on the Albrecht standard (Jeffery 1990). With respect to the inter-rater reliability research

question Low and Jeffery found that the consistency of FP counts “appears to be within the 30 percent reported by Rudolph” within organizations (Low and Jeffery 1990, p. 71).

Most recently, Kemerer conducted a large-scale field experiment to address, among other objectives, the question of inter-rater reliability using a different research design. Low and Jeffery chose a small group experiment, with each subject’s identical task being to count the FPs implied from the two program specifications. Due to this design choice, they were limited to choosing relatively small tasks, with the mean FP size of each program being 58 and 40 FPs, respectively. A possible concern with this design would be the external validity of the results obtained from the experiment in relation to real world systems. Typical medium sized application systems are generally an order of magnitude larger than the programs counted in the Low and Jeffery experiment (Emrick 1988) (Topper 1990). The Kemerer study tested inter-rater reliability using more than 100 different total counts in a data set with 27 actual commercial systems. Multiple raters were used to count the systems, whose average size was 450 FPs. The results of the study were that the FP counts from pairs of raters using a standard method¹ differed on average by approximately eleven percent. These results suggest that FPs are much more reliable than previously suspected, and therefore may indicate that wider acceptance and greater adoption of FPs as a software metric is appropriate.

However, these results also point out that variation is still present, and that the ideal goal of zero percentage variation has not been achieved in practice. In addition, this previous research, while identifying the magnitude of the variance, has not identified its sources. Therefore, of continued interest to managers are any systematic sources of this variation with accompanying recommendations for how to reduce or eliminate these variations.

¹As defined by International Function Points User Group Counting Practices Manual Release 3.0

3. RESEARCH METHODOLOGY

3.1 Introduction

This research was designed to address the question of the sources of decreased reliability of FP variations and consisted of two phases. In the first phase, key informants identified sixteen likely sources of variation. A survey of forty-five experienced users identified nine of these sixteen as especially problematic. In the second phase, detailed quantitative case study data on three commercial systems were collected and each system was counted using each rule variation. These cases are from three diverse organizations and management information systems.

3.2 Survey Phase

Development of the survey form was accomplished with significant involvement of the Counting Practices Committee (CPC) of the International Function Points Users Group (IFPUG). The committee consists of approximately a dozen experts drawn from within the membership of IFPUG. IFPUG consists of approximately 350 member organizations worldwide, with the vast majority being from the United States and Canada (Scates 1991). IFPUG is generally viewed as the lead organization involved with FP measurement and the CPC is the standards setting body within IFPUG (Albrecht 1990).

The CPC is responsible for the publication of the Counting Practices Manual (CPM), now in its third general release (Sprouls 1990). This is their definitive standards manual for the counting of FPs. In soliciting input from the CPC for this research, attention was focused on those systems areas for which (a) no current standard exists in the CPM, and (b) areas for which a standard exists but for which there is believed to be significant non-compliance.

From a series of meetings and correspondence with these key informants an original survey of fourteen questions was developed². This survey was pre-tested with members of the CPC and a small number of IFPUG member organizations not represented on the CPC, which resulted in the addition of two questions and some minor changes to existing questions. The final sixteen question survey is presented in Appendix A. This survey was mailed to eighty-four volunteer member organizations of IFPUG, who were asked to document how FP counting was actually done within their organization. No compensation was provided for completing the survey, although respondents were promised a summary of the results. Completion of the survey was estimated to require one hour of an experienced FP counter's time. Forty-five usable surveys were received, for a response rate of fifty-four percent. The survey respondents are believed to represent experienced to expert practice in current FP counting.

3.3. Case Study Phase

3.3.1 Introduction

While the survey phase of the research identified those areas that are likely sources of variation, it did not identify the magnitude of those effects. For example, while organizations may differ on the proper interpretation of a given FP construct, it may be the case that the situation described is relatively rare within actual information systems, such that differences in how it is treated may have negligible effect on an average FP count. Detailed data for each variant are required to assess the magnitude of the potential differences caused by each of the possible sources of variation. Given these data

²It is interesting to note that all of these questions deal with how to measure the five function count types, and none with the fourteen 'complexity factors'. This reflects the fact that any reliability concerns relating to the fourteen complexity factors are small, given that their potential impact on the final FP count is constrained by the mathematical formula [Albrecht and Gaffney, 1983] [Bock and Klepper, 1990]. This is in contrast to the five function types, where the impact of a different interpretation is unconstrained, and can be potentially very large. Empirical research has also documented the result that the impact of the fourteen complexity factors is small [Kemerer, 1987].

requirements, a quantitative case study methodology was chosen. As described by Swanson and Beath, this approach features the collection of multiple types of data, including documentation, archival records, and interviews (Swanson and Beath 1988).

The demand for detailed data with which to evaluate the multiple variations suggested by the surveys had two effects upon the research. First, a significant data collection and analysis effort was required for each case, since each variant required the collection of additional data and the development of a new FP count. Second, the detailed data requirements excluded a number of initially contacted organizations from participating in the final research.

The project selection criteria were that the projects were recently completed and for which there was an already completed FP count in the range of 200 - 600 FPs. This range was selected as encompassing medium sized application development and is the size range of the bulk of projects which are undertaken in North American systems development organizations today (Dreger 1989) (Kemerer 1991). None of them was composed of leading edge technology which might limit the applicability of standard FP analysis, such as "multi-media" or "compound document" systems. Rather, they represent typical MIS applications, and are described in more detail in the next section.

Obtaining the final usable three sets of case study data required the solicitation of ten organizations. Only these three possessed the necessary data and were willing to share these data with the researchers. These cases represent systems that are of the type for which FPs were developed, and which are representative of the type of systems developed and maintained by the original survey respondents.

The results were obtained using a variance analysis approach. Each of the systems submitted for the analysis had an original FP count and other relevant documentation. The analysis then systematically applied single variations of the counting rules which were identified in the research. These variations were those identified in the first phase

for further analysis because they were different from the CPM standard (or for which no standard had been established in the area), and they were being used by a significant population of the survey respondents.

3.3.2. Site A - Fortune 100 Manufacturer; Accounting Application

This case was provided by a large, diversified manufacturing and financial services company. This accounting application supports the need for rapid access to information from a variety of separate Accounts Payable applications. It was designed to operate in a PC/LAN environment, and is primarily used by accountants for inquiry purposes. It has built-in help facilities which can be maintained by the users of the system.

3.3.3. Site B - Fortune 50 Financial Services firm; MIS Data Base System

This case was provided by a large diversified financial services organization that has recently implemented a software measurement program. The system under study was developed as a stand-alone PC application, using a relational data base technology. The application is initially used by a single individual, but is expected to be expanded in its availability as its data bases become more robust. The application supports the management of the development function of the business, providing data and analysis to the managers of the software development and maintenance functions. The system was designed for ease of access, and has a robust set of menus to give the users easy access to the data.

3.3.4. Site C - Fortune 100 Manufacturing Company; Program Management System

This case was provided by the high technology division of a large aerospace manufacturing company. The system is used to track information concerning the management of various "programs" which are in process within the division. The system specifically tracks the backgrounds of the program managers. It was written in a fourth generation language, and

operates on a large central computer, which is accessible from networks of PCs and terminals. It has a simple menu structure, and contains no help capabilities.

4. RESULTS

4.1. Survey Results

Table 4.1.a contains the response data for the survey instrument in Appendix A. The number of possible responses varied by question from a low of three to a high of six. The table summarizes the percentage of survey respondents selecting each of the possible answers. In addition, the response which is compliant with the CPM is highlighted with a double-bordered cell. Given the extensive data collection and analysis requirements necessary to analyze each variant, the second phase of the research was designed to investigate only those topics identified by the survey as the most likely sources of variance. In order to determine which topics merited further attention in the case studies, a target minimum was set equal to a 50% compliant response rate, i.e., the topics selected as candidates for further study were those where more than 50% of the responses were different from the CPM standard. This cutoff, while arbitrary, was deemed appropriate given that these issues had been pre-selected as especially contentious³.

Therefore, the data in Table 4.1.a should be read as follows. The CPM standard answers (if existent) are placed in a double-bordered cell. If the percentage of answers within this 'target' cell is less than 50%, then the topic was regarded as a candidate for further study. For convenience, the maximum answer in each row is highlighted in bold and italics, as is the topic name. This allows an easy additional interpretation of the data, which is that questions for which the target answer is not the maximum answer (disregarding the 50%

³From Table 4.1.a it can be seen that the responses to two questions were near the cutoff point: number 7 with an agreement level of 49% and number 11 with an agreement level of 51%. To avoid *ex post* decision making with regard to the topics meriting further study, the original 50% guideline was strictly adhered to, with the result that question 7 was further investigated while question 11 was not.

cutoff) are those for which IFPUG needs to better communicate the standard to FP counters. For these data those questions are numbers 3, 4 and 5.

In the case of question 9 the CPM does not contain a counting standard for this issue, and thus no CPM compliant response is identified. For questions 13 and 14, the CPM does have a standard. Unfortunately, upon analysis of the survey data it was determined that the survey questions were sufficiently ambiguous as to not clearly differentiate a single correct answer. Therefore, no CPM "target" is shown for these two questions.

Table 4.1.a Phase I: Survey Results

Question Number and Subject	Percent by Response category						Candidate?
	1	2	3	4	5	Other	
1. Backup Files	2%	4%	40%	16%	27%	11%	Yes
2. Multi-function External Output Screens	29%	7%	42%			22%	Yes
3. Error Messages	14%	32%	21%			34%	Yes
4. Menu Function Types	37%	7%	2%	2%	51%	0%	Yes
5. Menu Function Count	37%	16%	5%	40%		2%	Yes
6. Help Messages Function Count	9%	7%	60%	11%		13%	No
6a. Help Messages Function Type	0%	30%	64%			7%	No
7. Help Screen Function Count	4%	7%	49%	31%		8%	Yes
7a. Help Screen Function Type	2%	0%	2%	23%	67%	5%	No
8. Report with Detail and Subtotals	89%	5%	5%			2%	No
9. Hard Coded Tables	30%	7%	52%			11%	Yes
10. Report with 2 selection criteria	59%	36%				5%	No
11. Report ordered with 2 criteria	44%	51%				4%	No
12. External Inquiry Function count weights	93%	2%				5%	No
13. Logical Internal File used as transactions for another system	38%	9%	33%			20%	Yes
14. External Interface File used as External Inputs for a system	36%	13%	11%	29%		11%	Yes

4.1.2. Questions not requiring further analysis

While discussions with the key informants of the standards setting committee suggested the sixteen survey questions as potential areas for variance, the results of the survey

showed that, for some questions, a majority of respondents were in compliance with the standards. Therefore, these results from these questions are only discussed here briefly, and were not the subject of the second phase of the research.

Responses to questions 8 and 12 were unique in their overwhelming adherence to the CPM. These questions were initially suggested by a definition of counting practices documented in a recent textbook (Dreger 1989). The results of the survey indicate that these variations in counting standards are not widely used.

There was acceptable levels of agreement among the respondents concerning questions 10 and 11, dealing with counting reports with multiple selection criteria and multiple sort sequences. The results of the survey were compliant with the CPM guidance as well. No case studies were developed for these variations.

Responses to questions 6 and 6a were also substantially in support of the CPM standard. These related to the counting of "Help Messages" which may appear on various screens. As the responses were largely compliant, they provided no significant interest in the study of counting variation. Responses to questions 7 and 7a also related to "Help Functions" but at the "Help Screen" level. There was less conformity as reflected by the response to question 7 at 49% compliance with the CPM, but the response to 7a showed strong agreement with standards. Therefore, question 7 was deemed to merit further study, but question 7a was not.

4.1.3. Questions that are candidates for further analysis

In the remaining nine questions (two with two possibilities each, for a total of eleven variants), there was significant variance from the CPM standards to warrant the further investigation of resulting potential variance from differing counting rule interpretations. These cases were identified by selecting the situations in which a majority of the

respondents identified the use of a counting rule which was different from the CPM standard, or for which no CPM standard exists.

Definition of the 11 variants

Variant 1: Counting Backup Files as Logical Internal Files - The CPM requires counting these files as Logical Internal Files, but only if they are specifically requested by the user due to legal or other business requirements. As Logical Internal Files have the highest weighting factors in function point counting, counting the backup file as a Logical Internal File could have a significant impact on the overall FP count.

Variant 2: Counting Backup Files as External Outputs - About twenty percent of the respondents to the survey indicated that they count backup files as External Outputs. The weighting factors for External Outputs are less than Logical Internal Files, but could have a significant impact on overall FP counts if there were a large number of such files.

Variant 3: Counting Add, Change, and Delete Outputs as separate functions - CPM counting rules allow the counting of each of the Add, Change and Delete transactions as a separate function type. However only forty-two percent of the respondents indicated compliance. Organizations which do not count these separately may lose up to 2/3 of the points from External Inputs, and somewhat less from External Outputs.

Variant 4: Counting Error Message Responses as individual data elements - Counting the data elements of a particular function type is necessary to determine the level of complexity for External Input transactions. Counting each error message response as a separate data element could force a Low or Average complexity function to be counted as Average or High complexity, increasing its FP value by up to 50%.

Variant 5: Counting Menus as an External Inquiry - CPM guidance is clear that navigational menus are not counted as individual function types, but their existence is a factor in increasing the FP complexity adjustment factor. Petitions to the CPC have

indicated that a) users see real value in menus, b) that systems are employing more and more menuing capability, and c) that creating menuing structures is consuming more development time. Variants 5 and 6 indicate alternate counting approaches which were in use by the survey respondents.

Variant 6: Counting Menus as one External Inquiry for each layer of menu - See Variant 5.

Variant 7: Counting Menus as one External Inquiry for each menu screen- See Variant 5.

Variant 8: Counting Help Screens as individual function types. –The CPM counting rules state that help screens are counted as External Inquiry function types, and that there is one External Inquiry type for each “calling screen.” In the survey, many of the respondents reported that they count one External Inquiry type for the entire suite of help capability, while others count each help screen combination as a separate External Inquiry Type. This variation could be significant in the overall count for a system with substantial help capabilities.

Variant 9: Counting “Hard Coded” data tables as Logical Internal Files. - The CPM does not currently have an official standard in this area. One view is that all files, whether “hard coded” or not should be counted as function types. Another view is that unless the files are “user maintainable” that they should not be counted. If there were sufficient numbers of “hard coded” tables, the FP count could be significantly affected, as Logical Internal Files are heavily weighted in FP counting.

Variant 10: Logical Internal File used as transactions for another system - This variant of rule interpretation and the following one had a great diversity of responses. Both have to do with the ways in which two systems interface with one another. One view is that files which are accessed for purposes other than just information reference purposes should be counted as transactions in one or the other system. The difficulty is centered around the

definition of the logical transaction (External Input or External Output) which is (or is not) taking place, and whether it should be counted in one system or the other.

Variant 11: Counting External Interface Files as External Inputs when used as transactions.

- See Variant 10.

Table 4.1.b below maps the eleven variants to the original survey questions.

Table 4.1.b: Case Study to Survey Question Mapping

Case Study Variants	Related Survey Questions
1: Counting Backup Files as Logical Internal Files	1. Backup Files
2: Counting Backup Files as External Outputs	1. Backup Files
3: Counting Add, Change, and Delete Outputs as separate functions	2. Multi-function External Output Screens
4: Counting Error Message Responses as individual data elements	3. Error Messages
5: Counting Menus as an External Inquiry	4. Menu Function Types
6: Counting Menus as one External Inquiry for each layer of menu	5. Menu Function Count
7: Counting Menus as one External Inquiry for each menu screen	5. Menu Function Count
8: Counting Help Screens as individual function types	7. Help Screen Function Count
9: Counting "Hard Coded" data tables as Logical Internal Files	9. Hard Coded Tables
10: Logical Internal File used as transactions for another system	13. Logical Internal File used as transactions for another system
11: Counting External Interface Files as External Inputs when used as transactions	14. External Interface File used as External Input transactions for a system

In each of these cases, the effect on the system FP count of each variation on the standard count was evaluated. It should be noted, however, that the total unadjusted FP count of an individual case could be affected by a combination of application of the rules, which might result in cumulative variations which exceed any one of the individual variances

from the application of a single rule change. A 'worst case' analysis will be presented after the presentation of the main results.

4.2. Case study Results

Each of the three cases is discussed individually below. For each of the cases, there are two analysis tables: one containing the base FP count (based on CPM 3.0), and one with a variance analysis summary. A summary of the results of all three cases appears in Table 4.2.4.a.

4.2.1. Site A results

The base size for the system analyzed at site A was 379 unadjusted FPs⁴. The system was a robust system with a wide range of function types developed under a relational data base technology. This system was developed with a high degree of interaction with the using organization. The users had an exceptionally high degree of interaction with the design and development team, and worked with them to develop and document the system. The documentation for this system was the most extensive of all the cases which were investigated. The functionality of the system does not demand a robust, multi-tiered menu system, but the users did require extensive "Help" capabilities. These capabilities allow the users to continue to update the "Help" screens as required by changes in business practice or better understanding of the assistance which the users of the system need. The error messages of the system were also highlighted using color and emphasized text. In the evaluation of complexity factors, the system rated high marks for its design for End User Efficiency.

⁴The original count (not the base count shown in Table 4.2.1.a) developed by site A was the only case which did not comply with all of the counting rules as contained in Release 3.0 of the CPM. The original count provided by the FP counters at site A was 418 FPs, which is 10% higher than the value achieved through application of the CPM. This is additional evidence of the need for this type of study, and for the further promulgation of counting standards.

Table 4.2.1.a: Base Count for Case A

Definition:	BASE FUNCTION POINT CLASSIFICATION			
	Low	Avg.	High	Total
External Input	11 x3= 33	0 x4= 0	0 x6= 0	33
External Inquiry	4 x3= 12	6 x4= 24	16 x6= 96	132
External Output	14 x4= 56	5 x5= 25	1 x7= 7	88
Log. Internal File	18 x7= 126	0 x10= 0	0 x15= 0	126
Ext. Interface File	0 x5= 0	0 x7= 0	0 x10= 0	0
Total Unadjusted Function Points:				379

This company began counting function points in about 1987, before the publication of CPM 3.0. They used an outside consultant for training the IS staff in the latest counting techniques, which was the best available source at the time. The variations in application of the counting rules may have been as a result of pre-CPM 3.0 recommendations.

The results of the FP variance analysis are presented in Table 4.2.1.b. Three of the variants (1, 2, and 11) produced significant variances in the count, where significant is a variance larger than the average eleven percent difference observed in previous research.

Three of the variants analysis require further explanation. Variant 3 shows a negative variance from the base count. This is a result of the particular counting rule, which allows the counting of Add/Change/Delete transactions as separate function types. Failure to apply the rule properly reduces the FP count. In all other variants, the rule prohibits counting particular function types. These variants result in positive variations from the base count.

Table 4.2.1.b Phase II: Case Study A Results

Variant	Site A	
	FP	% Δ
Base Count	379	
1. Backup Files as Logical Internal Files	484	28%
2. Backup Files as External Output Types	451	19%
3. Count Add/Chg/Del for External Output Types	355	-6%
4. Count Error Message Responses	379	0%
5. Count Each Menu Screen	391	3%
6. Count Each Layer of Menu Structure	385	2%
7. Count a suite of menus as one Query Type	382	1%
8. Count each separate Help Screen	403	6%
9. Count Hard Coded Tables as Logical Internal Files	dna*	-----
10. Logical Internal File used as transactions for another system	379	0%
11. Count External Interface Files as External Input Transactions	439	16%

Variant 9 is recorded as "dna" - data not available. No hard coded tables were noted in the documentation, but as there was no access to the source code to confirm this result it has been recorded conservatively.

The calculation for the impact of Variant 11 required making an assumption concerning the level of complexity of the transactions related to the "external interface files." The analysis assumes that the 15 associated transactions were of average complexity, resulting in a variance of 14%. The maximum impact (if they were of "high" complexity) would have been 21%, and the minimum impact (if they were of "low" complexity) would have been 11%.

4.2.2. Site B results

This system had an unadjusted FP count of 385 points, the largest of those studied. It was a well documented system, primarily used for management purposes within the Information Systems organization. The counts for the system were done manually and followed the CPM guidelines precisely.

Table 4.2.1.a: Base Count for Case B

Definition:	BASE FUNCTION POINT CLASSIFICATION							
	Low		Avg.		High		Total	
External Input	41	x3= 123	1	x4= 4	4	x6= 24	151	
External Inquiry	0	x3= 0	7	x4= 28	4	x6= 24	52	
External Output	1	x4= 4	10	x5= 50	1	x7= 7	61	
Log. Internal File	13	x7= 91	0	x10= 0	2	x15= 30	121	
Ext. Interface File	0	x5= 0	0	x7= 0	0	x10= 0	0	
Total Unadjusted Function Points:								385

This system was the most heavily “menued” of the systems studied, but did not have any “Help” capabilities. The system used extensive relational files, but did not have any External Interface Files.

Company B is implementing a quality improvement program within their information systems organization. A major part of that commitment is the measurement of various aspects of the systems development process. The subject system is the focal point of this measurement process. It is the data base for all the measurement data which is being collected within the several divisions of the company. The system was designed and implemented by the primary user, who happens to be a part of the IS community. The system was designed to operate in a stand-alone mode initially, with interfaces to other systems being developed in the future. Over time, the system will be expanded in scope so that the data base will be available for inquiry over a network. The menu suite would probably not be necessary if the system were to continue to be used by its designer. The design of the extensive menu capability reflects the recognition of needs of yet unidentified users.

This organization began its function point counting program in the mid 1980's and abandoned the measure due to a perceived lack of counting consistency. They have recently re-introduced the measure to the organization, waiting for the publication of CPM 3.0 before training the analysts in counting rules. The count for this system met all the CPM rule interpretations.

Table 4.2.2.b Phase II: Case Study "B" Results

Variants	Site B	
	FP	% Δ
Base Count	385	
1. Backup Files as Logical Internal Files	506	31%
2. Backup Files as External Output Types	451	17%
3. Count Add/Chg/Del for External Output Types	385	0%
4. Count Error Message Responses	385	0%
5. Count Each Menu Screen	427	11%
6. Count Each Layer of Menu Structure	394	2%
7. Count a suite of menus as one Query Type	388	1%
8. Count each separate Help Screen	385	0%
9. Count Hard Coded Tables as Logical Internal Files	dna*	----
10. Logical Internal File used as transactions for another system	385	0%
11. Count External Interface Files as External Input Transactions	385	0%

Variant 3 applies specifically to the existence of Add/Change/Delete *output* transactions. This system did not have separate transactions associated with outputs, therefore eliminating the count of any variation in the above table. As an aside, however, the subject system did have a wide range of A/C/D transactions associated with the External Inputs to the system, all of which were enumerated as individual function types. They are included in the base count. If these function type triples are counted only once, there would be a reduction of 96 FP, or 24%.

4.2.3. Site C results

This system was the smallest of the systems studied for the purposes of this research, with an unadjusted FP count of 208 FPs. This system was counted both manually, and then was checked using an expert system which verified both the computation of the FPs, but also applied the CPM counting rules consistently. The count of this system was fully compliant in every way with CPM guidelines.

This organization started its measurement program two years before they introduced FPs to measure size. Much data had been gathered within the organizations along a number of

dimensions. However, the ability to measure productivity was elusive since there was no consistently useful measure of system size. They implemented FPs just after the publication of CPM 3.0, learning the official interpretation of FP counting rules. Their original counts for this system were done manually applying the CPM rules. They then “audited” the count with the aid of the expert system. The system changed the original count in several cases based on particular rule interpretations.

Table 4.2.3.a: Base Count for Case C

Definition:	FUNCTION POINT CLASSIFICATION				
	Low	Avg.	High	Total	
External Input	17 x3= 51	2 x4= 8	0 x6= 0	59	
External Inquiry	16 x3= 48	8 x4= 32	0 x6= 0	80	
External Output	0 x4= 0	0 x5= 0	0 x7= 0	0	
Log. Internal File	9 x7= 63	0 x10= 0	0 x15= 0	63	
Ext. Interface File	1 x5= 5	0 x7= 0	0 x10= 0	5	
Total Unadjusted Function Points:				207	

This system had few menus, no specific External Outputs, but was dominated by External Inputs and inquiries. It was designed quickly to fulfill a very specific need, utilizing a fourth generation language and relational data base tool. This data base and the system were originally created for the responses to a “Program Manager Questionnaire” which was circulated in 1990. The system provides both pre-programmed inquiries and reports as well as the capability for ad-hoc inquiries. The documentation for this system was primarily the source code and the FP calculations. Since this was the only case study in which the authors had access to the source code, it was the only one in which a determination about “hard coded tables” could be made with certainty. The code revealed no “hard coded tables” which might have been counted as Logical Internal Files.

Table 4.2.3.2 Phase II: Case Study "C" Results

Site C		
Variants	FP	% Δ
Base Count	208	
1. Backup Files as Logical Internal Files	271	30%
2. Backup Files as External Output Types	244	17%
3. Count Add/Chg/Del for External Output Types	208	0%
4. Count Error Message Responses	208	0%
5. Count Each Menu Screen	214	3%
6. Count Each Layer of Menu Structure	214	3%
7. Count a suite of menus as one Query Type	208	0%
8. Count each separate Help Screen	208	0%
9. Count Hard Coded Tables as Logical Internal Files	208	0%
10. Logical Internal File used as transactions for another system	208	0%
11. Count External Interface Files as External Input Transactions	208	0%

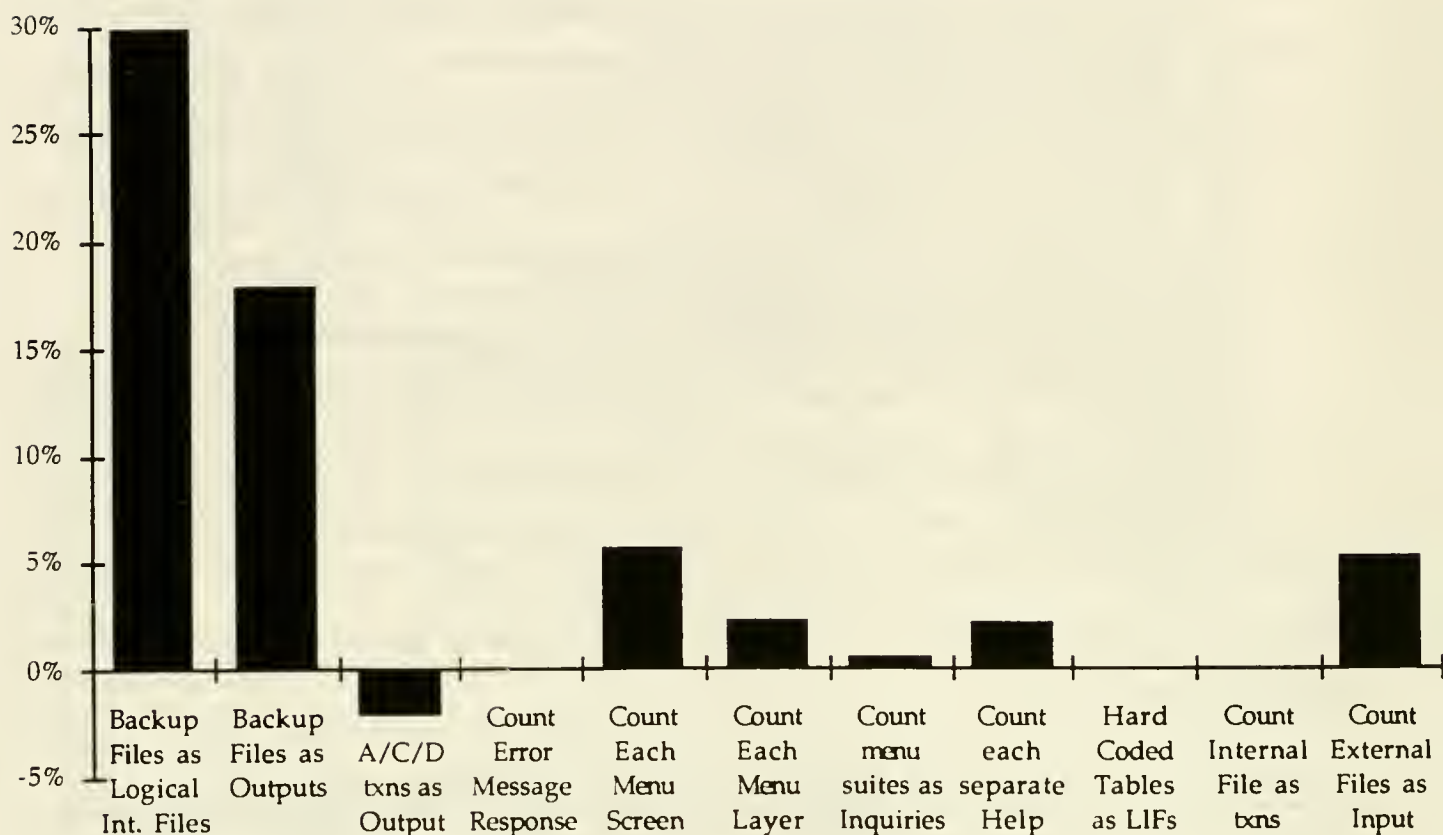
4.2.4. Summary of Results

Table 4.2.4.a summarizes the results for all three case studies. Figure 1 presents the average impact in graphic form.

Table 4.2.4.a Phase II: Case Study Results Summary

Cases	Site A		Site B		Site C		Mean
	FP	% Δ	FP	% Δ	FP	% Δ	% Δ
Base Count	379		385		208		
1. Backup Files as Logical Int. Files	484	28%	506	31%	271	30%	29.7%
2. Backup Files as External Output Types	451	19%	451	17%	244	17%	17.7%
3. Count A/C/D for External Output Types	355	-6%	385	0%	208	0%	-2.0%
4. Count Error Message Responses	379	0%	385	0%	208	0%	0%
5. Count Each Menu Screen	391	3%	427	11%	214	3%	5.7%
6. Count Each Menu Structure Layer	385	2%	394	2%	214	3%	2.3%
7. Count a suite of menus as one Query Type	382	1%	388	1%	208	0%	.7%
8. Count each separate Help Screen	403	6%	385	0%	208	0%	2.0%
9. Count Hard Coded Tables as Logical Internal Files	dna*	-----	dna*	-----	208	0%	0%
10. Logical Internal File used as transactions for another system	379	0%	385	0%	208	0%	0%
11. Count External Interface Files as External Input Transactions	439	16%	385	0%	208	0%	5.3%

Average Percentage Change in Function Point Count by Variant



4.2.4.1. Topics identified as consistent sources of significant variation

For a variant to be identified as a consistent source of significant variation it needed to generate more than a 10% difference in reliability in all three cases. Only one survey-identified variant met this criteria:

Counting File Backups - The most consistent variation in counts occurred in the area of counting backup files due, in part, to the fact that the Logical Internal Files have the highest individual FP counts. The impact of the differences in the counts was surprising. If backup files are counted, the cases identified an impact of 17% to 31% variation, the largest single source of variance. The lowest variability was observed in the case where

backup files were counted as External Output types and the highest in the case where they were counted as additional Logical Internal Files.

4.2.4.2. Topics identified as likely significant sources of variation

For a variant to be identified as a likely source of significant variation it needed to generate more than a 10% difference in reliability in at least one of the three cases. Two survey-identified variants met this criteria:

Counting Menus - In two of the cases, counting (or not counting) menus had an insignificant impact of the total FP count (3%). In one case, where the system was heavily supported by a robust set of menus the impact was more substantial (11%). This variation is sufficient to introduce a single source of variability which exceeds the typical variability of FP counts reported elsewhere, and is worth further analysis (Kemerer 1991).

One additional possibility is that as Graphical User Interfaces (GUI) become more widespread, users will demand more robust menuing capabilities. As this becomes the rule, rather than the exception, issues surrounding the counting of menus may become more significant in terms of the impact on reliability of FP counts.

Counting External Interface File Transactions - Two of the systems had interfaces to other systems. This situation was only observed in one of the systems studied. The other case (Site C) used an External Interface File strictly for reference purposes, and not to update a data base. The overall impact was below the threshold of 10%, but the single case in which it applied caused a 16% variation in count. The highest percentage of respondents to the survey (thirty-six percent) indicated that they would count the transactions. The IFPUG CPC has taken a clear position on counting these transactions, yet there is significant diversity in application of the rules. These results further indicate the need to communicate the counting rules and to reinforce the need for consistency.

4.2.4.3. Topics identified as possible significant sources of variation

The following variants resulted in 5% or greater variance in at least one case:

Counting Add/Change/Delete Transactions - The question stated in the survey focused on the counting of External Outputs from A/C/D transactions. Only one of the case study examples identified individual outputs from the A/C/D transaction sources. In this case, there was a variance in the total count of 6%. In two of the cases, the FP counts included separate counts of A/C/D *input* transactions. This is compliant with CPM guidance. However, if there were only one External Input function counted for each of the A/C/D triples, there would have been a 25% reduction in overall FP counting one case, and a 10% reduction in the other. Again, these are substantial variations in the overall FP counts, which could have a significant detrimental impact on reliability.

Counting Help Screens - Only one of the systems contained a "Help Facility." In the case of that one system, changes in the application of the counting rules resulted in a six percent overall shift in the FP count. This variation, while smaller than the impact of backup files, is still a significant percentage of the average variability. Users are increasingly requiring internally built systems to match the functionality of off-the-shelf software, which is typically equipped with Help and other facilities. It is reasonable to expect that these functions will account for more of the overall functionality of systems in the future. In this regard, a current six percent variation due to this rule interpretation is one which may demand further consideration.

4.2.4.4. Topics identified as consistent non-sources of significant variation

Other survey-identified variants tended to result in small or zero bottom line variances:

Counting Error Message Responses -None of the cases studied had error messages associated with External Input transactions. This is the only case that CPM 3.0 allows the counting of error messages. In the one case (Site A) in which the error messages were

present, they were only associated with inquiries. Even if the counting rule were to be applied to the inquiries there was very little variation. Of the ten transactions (inquiries) which were potentially affected, most were already classified as High complexity. These inquiries already had achieved the highest point value available, and counting any additional data elements could not have raised the point score. Only three of these transactions, which were classified as Average could have been affected in a recount. The analysis would have increased their point value from 4 to 6 points each, increasing the total FP count for the system by 6 points or one percent. While this observation does not result in any additional point counts, it is indicative of the small impact to be expected through this variant.

Counting Menu Screens (and other variants) - There were three variants analyzed for their impact on the overall count. Only one of these (counting each screen) had the potential of making a substantial impact on the overall FP count. The mean impact was less than 6% across the three cases, but one case registered an 11% change in overall count as a result of counting the menu suite. This could be significant for two reasons: 1) Users are demanding more heavily menued systems now than in the past, and 2) 40% of the respondents to the survey indicated that they would count all the screens as inquiries. With the combination of these two factors, there is a need to publicize the CPM rules to improve compliance.

Counting Hard Coded Tables - The source code necessary to investigate this feature was only available at site "C" where it was determined that no hard coded tables existed, and hence the impact of counting variants was zero. Clearly, this result should be interpreted especially cautiously, since it may be an artifact of this particular site.

Counting Files used by Other Systems as Transactions - None of the three cases which were reviewed contained citations of Internal Logical Files which were used by other systems as Input Types. The case studies were restricted to single systems, and were all recently

developed. It is possible that one or all of these systems may have an Internal Logical File used as an External Input to another system in the future. The rule may be tested at that time, but was not tested in these cases.

4.2.4.5. 'Worst case' analysis

In all of the above analysis each variant was analyzed separately in order to identify those variants that most merited management attention. An additional question could be asked, which is, what if a site were to be unfortunate enough to have chosen every variant that would maximize the difference between its FP count and the count achieved by following standard practice. Note that this difference is not simply the sum of the eleven variants, as not all of the variants are independent. For example, variants 1 and 2 are two different means of treating backup files. A site could choose one or the other instead of the standard, but could not logically choose both. In particular, the maximum positive variance scores shown in Table 4.2.4.1 are the summation of the percentage variance from variants 1, 5, 8 and 11.

Table 4.2.4.1 Worst Case Results

Site	Maximum Negative Variance	Maximum Positive Variance
A	-6%	53%
B	0%	42%
C	0%	33%
Average	-2%	43%

It should be emphasized that the average worst case result of 43% is not inconsistent with previous research reporting variance among counters of approximately 11%. This is because the previous research focused on typical or average case behavior, whereas the larger figure represents the largest possible variance at these sites given the choice of those variants previously identified as contentious areas and deliberately choosing those which would create the largest arithmetic difference.

4.3 Summary of Results

4.3.1 General Results

In general, the broad message to be taken away from this research is that FPs are highly reliable in practice. This conclusion is the result of the relatively small size of almost all of the variances demonstrated in case studies that were deliberately designed to investigate areas that were believed, *a priori*, to be significant sources of variance. These results should be encouraging both to organizations that have already adopted FPs, and for organizations that are currently considering their adoption.

Beyond this general result, however, there are clearly areas in which the definition of FPs could be improved. Most important among these is the proper counting of backup files. IFPUG needs to adopt and promulgate a clear and consistent standard on this topic, as this is the area that was identified in the research as posing the greatest threat to counting reliability.

4.3.2 Implications for standards settings

There is a need to act on the findings of this research. Standard setting bodies such as the IFPUG CPC should take a series of actions to improve the reliability of FP counts. These are:

- ***Identify and resolve outstanding and contentious issues*** - Even after the specific issues addressed in this research are resolved, the rapid pace of change in information technology virtually guarantees that new issues will arise. To address this issue, a regular approach by a standards setting body needs to be put into place to institutionalize the type of research presented here. This research would consist of two phases, the first an identification phase to identify potential problem areas, and a case study phase where the effect of these potential problems is assessed. Without such a process in place it is likely that FP counting standards are likely to significantly lag actual practice.
- ***Communicate standards for issues of frequent variation*** - A special communication should be prepared to emphasize the need for consistent application of existing counting rules. This conclusion is underscored by the non-compliance results shown in the survey.

- *Continue research into areas of potential variability-* There are other areas of variability which will become more prominent in the future. There must be a continuing program of research to insure that these areas are identified and counting standards written.

The need for greater communication of existing standards is readily apparent from the data in Table 4.1.a. The results of a survey of leading FP measurers demonstrate that for three issues, Error Messages, Menu Function Types, and Menu Function Count the majority answer was not the CPM standard. This indicates a need for greater communication of the CPM results to the membership⁵. The survey also revealed issues, such as External Inquiry function weighting, for which no additional special effort is deemed necessary.

4.3.3 Implications for automation of FPs

A critical precursor to the successful automation of FP counting through either stand-alone tools or embedded within CASE technology is the clear definition of measurement conventions. The current research results have three implications for the automation of FP counting. The first is the obvious need for the tools to carefully define their counting conventions, given the potential impact of adopting non-standard variants. Second, the tools should clearly communicate these conventions to the user. Failure to do so may lead to unsuccessful adoption of the tool by organizations that have previously been counting FPs manually. If, for example, a tool has adopted radically different conventions than those used at the site, then initial benchmarking of the tool by experienced users may come to the conclusion that the tool is inaccurate, when, in fact, it may be merely consistently applying variant counting conventions. Finally, a suggestion for tool vendors arising from these results is to provide some sensitivity analysis as part of the output of the tool. For example, following the variance approach taken in this research, the tool could produce as output both its standard count plus some alternative counts based on differing

⁵Since this survey was completed, the CPC has published CPM release 3.1, and is expected to publish the 3.2 update in the Fall of 1991.

assumptions. This could also highlight for users which features of the application are most significant in driving the final count, which might be a useful planning tool for project managers.

4.3.4 Implications for organizations counting FPs

Consistent counting of FP within an organization is of extreme importance. It provides the basis for comparison of systems measures across system, departments, and locations. This consistency can be gained by creating one's own standards, or by adopting the standards of others. The results of the research and the case studies indicate that organizations which adopt the CPM 3.0 standards do count reliably. Its adoption can provide a quick basis for the movement to consistency, and like all industry standards, will be updated to reflect contemporary issues in counting.

In both cases where the organizations were trained using the CPM 3.0, the base count was in compliance with the counting practices. In the case where the organization had been trained in counting FP before the publication of CPM 3.0 there were significant deviations from the CPM.

Measurement is the means by which management knows that objectives are being met. The accuracy of these measures over time, and across various systems, organizations and even companies is an essential component to appropriate decision making. Through this and related research Function Points have been shown to be a reliable measurement instrument. Managers should adopt them as a measure of system size, and follow and endorsed standard in their use. Function Points are the only measure supported by an independent standards setting body, with an established problem resolution process. It is this standard setting function which will continuously improve the ability of FP to measure system size. This improvement requires the active support of organizations which are using FP-based measures in identifying potential sources of variation, and suggesting solutions to the standard setting body.

5. CONCLUDING REMARKS

This paper identifies the source and impact of variations in the application of FP counting rules. The results of this analysis should provide guidance to FP standard setting bodies in their deliberations upon rule clarification, and to practitioners as to where the difficulties lie in the current implementation of FPs. The result of this effort should continue the process of improving the quality and reliability of measures of software size, productivity and quality.

Improving the quality of this one measure is but a start in the effort to improve management's ability to measure all the aspects of software development and maintenance. Objectives of managers today include productivity and quality, but are certainly not limited to them. Increased efforts to improve the reliability of these measures will continue to enhance their acceptance and credibility in both the worlds of the systems professionals and general management.

The issues upon which this research have focused center on the clarification of counting guidelines for systems which are "traditional" in nature. The object is to refine the counting guidelines, and to drive out the ambiguity of current measurement conventions. This is a relevant and important issue, since there are so many systems for which these measures are relevant.

However, the issue of measurement reliability is much larger than just the issues outlined within the context of this text. The advent of event driven, object oriented systems; knowledge based systems; real-time and scientific systems may require re-definition of FPs or the development of one or several new measures to identify system size. For example, an initial set of metrics for object-oriented design has been proposed (Chidamber and Kemerer 1991).

FPS currently provide the only established industry standard of size measurement in the area of systems development. The measurement of efficiency requires equivalent standardization of resource (cost and time) measurement. Few organizations have the same rules for accounting for staff time applied to projects. It is probably fair to say that no two organizations account for costs in the same way. If there is to be further comparison of measurement across companies, and the development of more refined estimating capabilities, standards will need to be established in a wide variety of areas of software development management. Some recent work by the IEEE Software Productivity Metrics Working Group of the Software Engineering Standards Subcommittee is a step in this direction (IEEE 1990).

Systems development is an intellectual activity, the conversion of an idea into software. However, if the IS profession is to improve the way in which this critical work is done then measurement of this intellectual activity is necessary. Perfect measures may never be developed, but efforts directed toward this goal should result in improved metrics and therefore wider adoption in practice.

The interest expressed in the area of measurement is growing. More people believe that the activity can be effectively measured and managed and further development of measurement standards is to be encouraged.

APPENDIX A

FP Counting Practices Survey

In this section, we would like you to answer the questions using your organization's Function Point counting conventions.

1. How does your site count backup files? (check one of the following):

- ☐ Always count them as Logical Internal Files
☐ Always count them as External Outputs
☐ Count them as Logical Internal Files, but only when backup files are requested by the user and/or auditors
☐ Count them as External Outputs, but only when backup files are requested by the user and/or auditors
☐ Never count them
☐ Other (Please explain): _____

2. Please refer to the following screen example titled "Multi-Function Address Screen".

How many unique External Outputs would your site consider this screen to indicate?

Assume that a successful transaction is indicated by displaying a confirmation message on this screen. (check one of the following):

- ☐ One, because the output processing is the same for add, change, and delete functions.
☐ Two, because the output processing for the add and change are the same, but the output processing for the delete is different.
☐ Three, because add, change, and delete indicate three distinct outputs.
☐ Other. (Please explain): _____

Multi-Function Address Screen

Name: _____

Address: _____

City: _____

State: __ Zip _____

transaction confirmation message goes here

PF1 = Add PF2 = Change PF3 = Delete

3. Please refer to the following screen example titled "Add an Address Screen - I".

Assuming two files are referenced, what complexity would your site assign to the External Output associated with this screen? (check one of the following):

- ☐ Low. There are five data elements because error messages are not counted.

_____ Average. There are six data elements because error messages get counted only once as only one message appears on the screen.

_____ High. There are 25 data elements because each possible error message is counted as an element. Other. Please explain:

Add an Address Screen - I

Name: _____

Address: _____

City: _____

State: __ Zip ____

error message goes here

All Possible Error Messages (20 in total)

1. Name too long.
2. Name too short.
3. Not a valid city.
4. Not a valid state.
- ... etc....
- ... etc....
- 19 Zip code must be numeric.
20. Wrong # digits in zip code.

4. Please refer to the following screen Layout Hierarchy, consisting only of a main menu and five sub-menus, what Function Type(s) would your site use in counting these menus? (check as many as apply):

- _____ Not applicable - menus are not counted
- _____ External Input
- _____ External Output
- _____ Logical Internal File
- _____ External Inquiry
- _____ External Interface

Screen Layout Hierarchy

```

Main Menu  ---- |---- Manage Inventory ----
                |---- Plan Acquisition -----
                |---- Update Catalogue -----
                |---- Support Inquiries -----
                |---- Produce Reports -----
  
```

5. Referring again to the Screen Layout Hierarchy, how many functions would your site count based on this hierarchy? (check one of the following):

- _____ 0, because menus are not counted
 _____ 1, because menus only get counted once regardless of the number of screens
 _____ 2, because there are two levels
 _____ 6, because there are six menu screens
 _____ Other. Please explain: _____

6. Please refer to the following screen example titled "Add an Address Screen- II". Based on this screen, how many additional functions would your site count due to the help messages? The help message displayed varies depending on the field the cursor is on. (check one of the following):

- _____ 0, but the complexity rating would reflect the presence of help messages
 _____ 0, but the General Systems Characteristics adjustment would reflect the presence of help messages
 _____ 1, because all help messages are treated as a single function
 _____ 5, because there are 5 help messages
 _____ Other. (Please explain): _____

Add an Address Screen -II

Name: _____
 Address: _____
 City: _____
 State: __ Zip__

help message goes here

Help Messages

1. Type last name, first name.
2. Address can only be one line.
3. Type name of city.
4. Type 2 character state code.
5. Type 5 or 9 digit zip code.

6a. Referring to the help messages of question 6, how would your site classify the function type for the messages? (check one of the following):

- ☐ External Input
☐ External Outputs
☐ External Inquiries
☐ Other. (Please explain): _____

7. Given the data entry screen of question 6, if there was one help screen per field (rather than a help message per field), how many additional functions would your site count due to the help screens? (check one of the following):

- ☐ 0, but the complexity rating would reflect the presence of help screens
☐ 0, but the General Systems Characteristics adjustment would reflect the presence of help screens
☐ 1, because all help screens are treated as a single function
☐ 5, because there are 5 help screens
☐ Other. (Please explain): _____

7a. Referring to the help screens of question 7, how would your site classify the function type for the screens? (check one of the following):

- ☐ Internal Logical Files
☐ External Interface Files
☐ External Input
☐ External Outputs
☐ External Inquiries
☐ Other. (Please explain): _____

8. Assume a report with detail lines, subtotals, and a grand total, where all lines have the same format. At your site, would you count this as:

- ☐ One External Output, with the subtotals and grand totals adding to the number of data elements.
☐ Two External Outputs: one including only the detail lines, and another including only the subtotals and grand totals.
☐ Three External Outputs: one including only the detail lines, another including only the subtotals, and another including only the grand totals.
☐ Other. (Please explain): _____

9. What function type does your site use for hard coded tables (i.e. tables which only a programmer, and not an end-user can change)? (check one of the following):

- ☐ Logical Internal Files, because they are files
☐ External Interfaces None, because they are not user-changeable
☐ Other. (Please explain): _____

10. Please refer to the following report layout titled Customer Orders. Assume that this report can be produced with either of two selection criteria: by selecting dates or by selecting customer numbers. The dates is ordered (sorted) by customer number regardless of the selection criteria used. How many External Outputs would your site count this report as? (check one of the following):

- _____ One, because the report format is the same for both selection criteria
 _____ Two, because the data is different depending on the selection criteria
 _____ Other. (Please explain): _____

Customer Orders

Cust #	Part #	Order Date	Quantity
1111	1111	1/1/88	11
2222	2222	2/2/89	22
3333	3333	3/3/89	33

11. Referring again to the report layout titled "Customer Orders". Assume that this report can be ordered (sorted) with either of two criteria: by date or by customer numbers. How many external outputs would your site count this report as? (check one of the following):

- _____ One, because the report format is the same for both ordering criteria
 _____ Two, because the data is different depending on the ordering criteria
 _____ Other. (Please explain): _____

12. For External Inquiries, which of the following sets of function point weights does your site use for low, average, and high complexity? (check one of the following):

- _____ Three for Simple, Four for Average, Six for Complex
 _____ Four for Simple, Five for Average, Six or Seven for Complex
 _____ Other. Please describe: __Simple, __Average, __Complex

13. If Application A reads one of Application B's Logical Internal Files and converts the data into transactions to update one of its own Logical Internal Files, how would your site classify the Logical Internal File in Application B? (check one of the following):

- _____ As a Logical Internal File and an External Interface File
 _____ As a Logical Internal File and an External Output File
 _____ Only as a logical Internal File
 _____ Other. (Please explain): _____

14 If Application A creates a file of transaction data from Application B's Logical Internal File, how would your site classify Application A's transaction file? (check one of the following):

- ☐ As an External Input
- ☐ As an External Interface File
- ☐ As a Logical Internal File
- ☐ As nothing (i.e. it would not be counted), because it is a temporary file.
- ☐ Other (Please explain):

BIBLIOGRAPHY

- Albrecht, A. J., "Foreword", in *IFPUG Function Point Counting Practices Manual*, Sprouls, J. (ed.), International Function Point Users Group, Westerville, OH, (1990).
- Albrecht, A. J. and J. Gaffney, "Software Function, Source lines of Code, and Development Effort Prediction: A Software Science Validation", *IEEE Transactions on Software Engineering*, SE-9 (6): 639-648, (1983).
- Banker, R. D., S. M. Datar and C. F. Kemerer, "A Model to Evaluate Variables Impacting Productivity on Software Maintenance Projects", *Management Science*, 37 (1): 1-18, (January 1991).
- Banker, R. D. and C. F. Kemerer, "Scale Economies in New Software Development", *IEEE Transactions on Software Engineering*, SE-15 (10): 416-429, (October 1989).
- Behrens, C. A., "Measuring the Productivity of Computer Systems Development Activities with Function Points", *IEEE Transactions on Software Engineering*, SE-9 (6): 648-652, (November 1983).
- Bock, D. B. and R. Klepper, "FP-S: A Simplified Function Point Counting Method", Southern Illinois University at Edwardsville Working Paper (October 12, 1990).
- Boehm, B., *Software Engineering Economics*, Prentice-Hall, Englewood Cliffs, NJ, (1981).
- Boehm, B., "Improving Software Productivity", *Computer*, 20 (9): 43-57, (September 1987).
- Carmines, E. G. and R. A. Zeller, *Reliability and Validity Assessment*, Sage Publications, Beverly Hills, CA, (1979).
- Chidamber, S. R. and C. F. Kemerer, "Towards a Metrics Suite for Object Oriented Design", *6th ACM Conference on Object Oriented Programming, Systems, Languages and Applications (OOPSLA)*, Phoenix, AZ, October, 1991,
- Cooprider, J. and J. Henderson, "A Multi-Dimensional Approach to Performance Evaluation for I/S Development", MIT Center for Information Systems Research Working Paper number 197, (December 1989).
- Desharnais, J.-M., *Analyse statistique de la productivite des projets de developpement en informatique a partir de la technique des points de fonction (English version)*, Universite du Quebec a Montreal Masters thesis, (1988).
- Dreger, J. B., *Function Point Analysis*, Prentice Hall, Englewood Cliffs, NJ, (1989).

Bibliography

Emrick, R. D., "Software Development Productivity Second Industry Study", *International Function Point Users Group Spring Conference Proceedings*, Dallas, Texas, May 16-19, 1988, pp. 1-44.

IEEE, "Standard for Software Productivity Metrics", Institute of Electrical and Electronics Engineering Software Productivity Metrics Working Group of the Software Engineering Standards Subcommittee Draft report P1045-D2.1, (March 29, 1990).

Jeffery, R., personal correspondence, August 15, 1990.

Jones, C., *Programming Productivity*, McGraw-Hill, New York, (1986).

Jones, C., "A Short History of Function Points and Feature Points", Software Productivity Research, Inc. mimeo version 2.0, (February 20 1988).

Kemerer, C. F., "An Empirical Validation of Software Cost Estimation Models", *Communications of the ACM*, 30 (5): 416-429, (May 1987).

Kemerer, C. F., "Reliability of Function Points Measurement: A Field Experiment", MIT Sloan School Working Paper 3193-90-MSA, (January 1991).

Low, G. C. and D. R. Jeffery, "Function Points in the Estimation and Evaluation of the Software Process", *IEEE Transactions on Software Engineering*, 16 (1): 64-71, (January 1990).

Perry, W. E., "The Best Measures for Measuring Data Processing Quality and Productivity", Quality Assurance Institute Technical Report (1986).

Pressman, R. S., *Software Engineering: A Practitioner's Approach*, 2nd ed., McGraw-Hill, (1987).

Reifer, D. J., "Asset-R: A Function Point Sizing Tool for Scientific and Real-Time Systems", *Journal of Systems and Software*, 11 159-171, (1990).

Rudolph, E. E., "Productivity in Computer Application Development", University of Auckland, Dept. of Management Studies, Auckland, New Zealand Working Paper 9, ISSN 0110-4756, (March 1983).

Scates, C., personal communication, August 5, 1991.

Sprouls, J. (ed.), *IFPUG Function Point Counting Practices Manual*, Release 3.0 Ed., International Function Point Users Group, Westerville, OH, (1990).

Swanson, E. B. and C. M. Beath, "The Use of Case Study Data in Software Management Research", *Journal of Systems and Software*, 8 63-71, (1988).

Symons, C. R., "Function Point Analysis: Difficulties and Improvements", *IEEE Transactions of Software Engineering*, 14 (1): 2-11, (January 1988).

Bibliography

Topper, A., "CASE: A Peek at commercial developers uncovers some clues to the mystery", *Computerworld*, XXIV (15): 61-64, (April 9, 1990).

Whitmire, S., S. Galea, R. Gold, J. Basic, M. Crow and J. Tollar, "Scientific and Real-Time Software Size Metrics - Draft Preliminary Technical Report", The Boeing Company technical report (January 9 1991).

Zwanzig, K. (ed.), *Handbook for Estimating Using Function Points*, GUIDE International, Project DP-1234, (1984).

MIT LIBRARIES DUPL



3 9080 00748710 8

Date Due

Lib-26-67

